

Maskinorienterad programmering

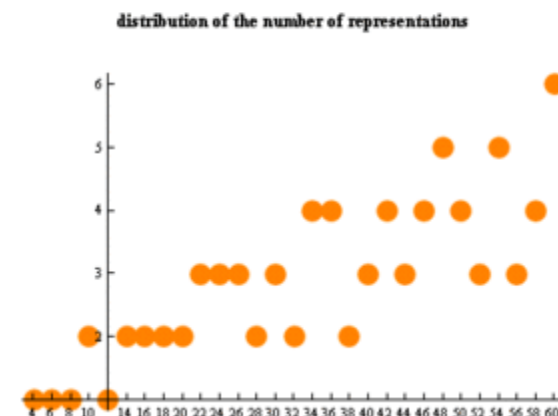
De första versionerna av denna uppgift konstruerades av Jan Skansholm.

Goldbach hypotes

Goldbach's conjecture is one of the oldest unsolved problems in number theory and in all of mathematics. It states: Every even integer greater than 2 can be expressed as the sum of two primes.

Such a number is called a Goldbach number. Expressing a given even number as a sum of two primes is called a Goldbach partition of the number.

(52 = 5 + 47, 52 = 11 + 41, 52 = 23 + 29)
(54 = 7 + 47, 54 = 11 + 43, 54 = 13 + 41, 54 = 17 + 37, 54 = 23 + 31)
(56 = 3 + 53, 56 = 13 + 43, 56 = 19 + 37)
(58 = 5 + 53, 58 = 11 + 47, 58 = 17 + 41, 58 = 29 + 29)
(60 = 7 + 53, 60 = 13 + 47, 60 = 17 + 43, 60 = 19 + 41, 60 = 23 + 37,



The number of ways an even number can be represented
as the sum of two primes

(Wikipedia)

Här använder du Codelite och GCC för värddatorsystemet.

Uppgift 1: Primtal

Ett primtal är ett heltal >1 som inte är jämnt delbart med några andra tal än med 1 och med sig själv.

Uppgiften är att skriva ett C-program som läser in positiva heltal från tangentbordet och som undersöker om talen är primtal eller inte. Programmet består av två funktioner:

```
int main(int argc, char **argv)
int is_prime(int n);
```

Funktionen `is_prime` skall undersöka om ett parametern `n` är ett primtal. Som resultat skall funktionen returnera värdet 1 om det talet `n` är ett primtal och värdet 0 annars. För att undersöka om ett tal är ett primtal kan man direkt använda definitionen av primtal. Man undersöker alltså om någon division av `n`, med något annat tal `m` ($m < n$, $m > 1$) ger ett resultat utan rest. Om divisionen inte går jämt ut för något sådant tal `m`, måste `n` vara ett primtal.

- Skapa ett nytt projekt `is_prime`.
- Implementera funktionen `int is_prime(int n)`
- Använd huvudprogrammet till höger för att testa funktionen. Funktionen `main` läser in ett tal i taget från tangentbordet. För varje tal större än 1 anropas funktionen `is_prime` för att undersöka om det inlästa talet är ett primtal eller inte. Resultatet av testet skrivs sedan ut till konsollfönstret
- Kontrollera programmets funktion med *CodeLite*. förvissa dig om att `is_prime` fungerar som avsett innan du går vidare till nästa uppgift

```
int main(int argc, char **argv)
{
    int x;
    printf("Is prime test\n");
    while(1)
    {
        printf("Enter number to check:");
        scanf("%d",&x);
        if( x < 2 ) break;
        if( is_prime( x ) )
            printf("%d is a prime number\n", x);
        else
            printf("Nop, %d is NOT a prime number\n", x);
    }
    return 0;
}
```

Uppgift 2: Hypotesen

Goldbach hypotes säger att varje jämnt naturligt tal (utom 2) kan skrivas som summan av två primtal. Hypotesen ställdes upp i mitten av 1700-talet och anses allmänt vara sann. Den är dock ännu ej bevisad.

Uppgiften är nu att skriva ett C-program som upprepade gånger frågar efter ett tal från tangentbordet. Om talet är jämnt skall programmet skriva ut alla möjliga kombinationer av två primtal som bildar talets summa. Varje kombination av primtal skall endast skrivas ut en gång.

Ett exempel på hur det kan se ut när man kör programmet visas till höger:

Om ett inläst tal är udda skall programmet ge en felutskrift. Körningen skall avslutas om man matar in ett tal som är mindre än eller lika med 2.

Om t.ex. talet 14 läses in, så skall programmet endast ge ett av primtalsparen (3,11) och (11,3) som utdata.

```
Ge ett tal. Avsluta med ett tal <= 2: 14
3+11
7+7
Ge ett tal. Avsluta med ett tal <= 2: 100
3+97
11+89
17+83
29+71
41+59
47+53
Ge ett tal. Avsluta med ett tal <= 2: 897
Talet är udda!
Ge ett tal. Avsluta med ett tal <= 2: 1
```

Programmet har tre funktioner: `is_prime`, `goldbach` och `main`.

```
int is_prime(int n);
```

samma funktion som i uppgift 1.

```
int goldbach( int start, int sum);
```

Funktionen letar efter första talpar (x och sum) där det gäller att: $x + (sum - x) = sum$ och att x respektive $(sum - x)$ båda är primtal. Iterationen börjar med $start$ och pågår till ett sådant talpar hittas, då returneras x , eller till alla möjligheter undersökts, i detta fall returneras 0.

```
int main(int argc, char **argv)
```

I `main` skall all inläsning och alla utskrifter ske. När ett tal n har lästs in kontrolleras först att n uppfyller villkoren dvs. är jämnt och större än 2. Därefter undersöks alla kombinationer av talen n_1 och n_2 sådana att $n_1 + n_2 = n$. Om både n_1 och n_2 är primtal ska detta talpar skrivas ut som $n_1 + n_2$.

- Skapa ett nytt projekt `goldbach`.
- Implementera även funktionerna:


```
int goldbach(int start, int sum);
int main(int argc, char **argv);
```
- Kontrollera programmets funktion med *CodeLite*.